

Minitutorial Github

Para la colaboración Janus

December 4, 2023

1 Paso 1. Sign in

A partir de la página oficial de Github <https://github.com> pulsamos en *Sign in* e introducimos correo (janus2collaboration@outlook.com) y contraseña.

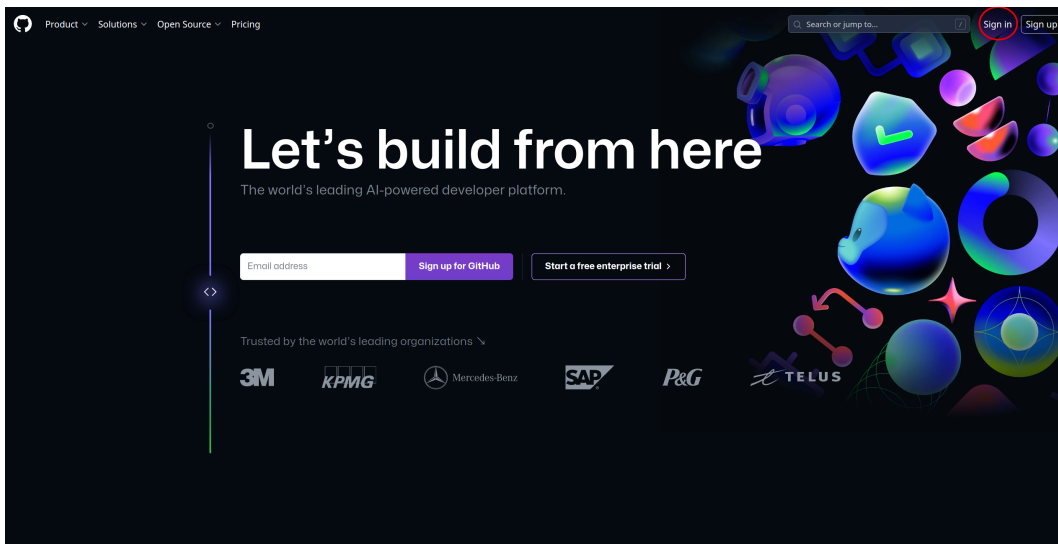


Figure 1: Interfaz de registro.

2 Paso 2. Crear repositorio

Una vez hagamos el registro, es realmente sencillo crear un repositorio. A partir de la página principal (Fig. 2) pulsamos sobre nuestro perfil y vamos a nuestros repositorios (Fig. 3).

Una vez en la página de nuestros repositorios (Fig. 4) podemos ver los que tenemos anteriormente (en nuestro caso el de caosdin) y podemos crear uno nuevo.

Si decidimos crear uno nuevo, se abre una página nueva en la que tenemos que elegir varios parámetros de nuestro repositorio (Fig. 5).

Lo más importante es el nombre del repositorio, que sea público o privado (en nuestro caso, queremos que cualquiera pueda acceder), añadir un README, y la licencia.

El fichero README.md es muy importante porque lo que haya escrito dentro se abre directamente bajo el repositorio y se suele usar como "guía" para saber cómo están organizados los datos, programas, figuras, ... del repositorio en cuestión.

Por otro lado, la licencia que usé la otra vez es la del MIT. Es una licencia tipo copyleft muy enfocada al uso de software y la usé porque es relativamente estándar, pero se pueden usar otros tipos de licencias dependiendo de cómo queramos que la gente que acceda al repositorio pueda usar los datos.

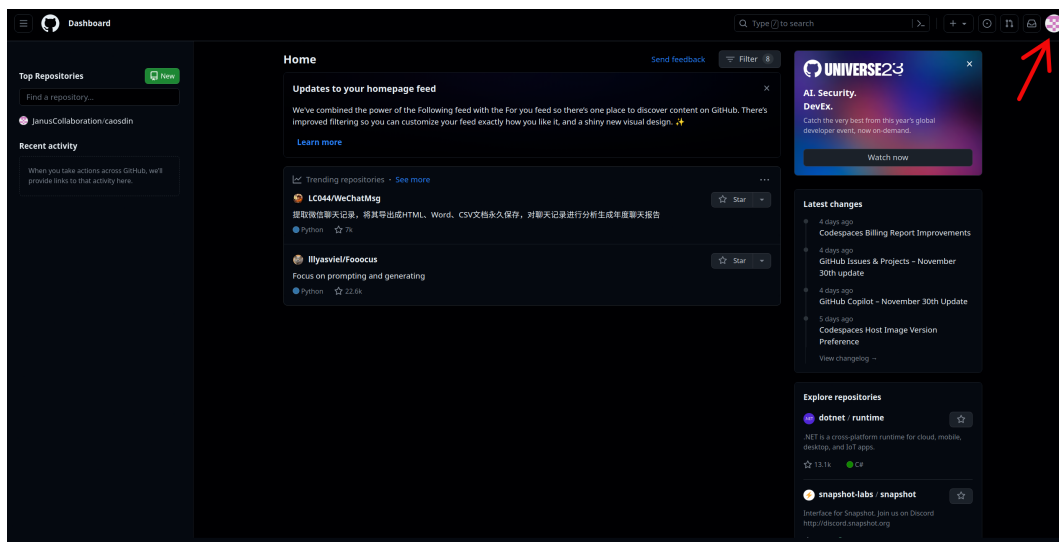


Figure 2: Página de inicio.

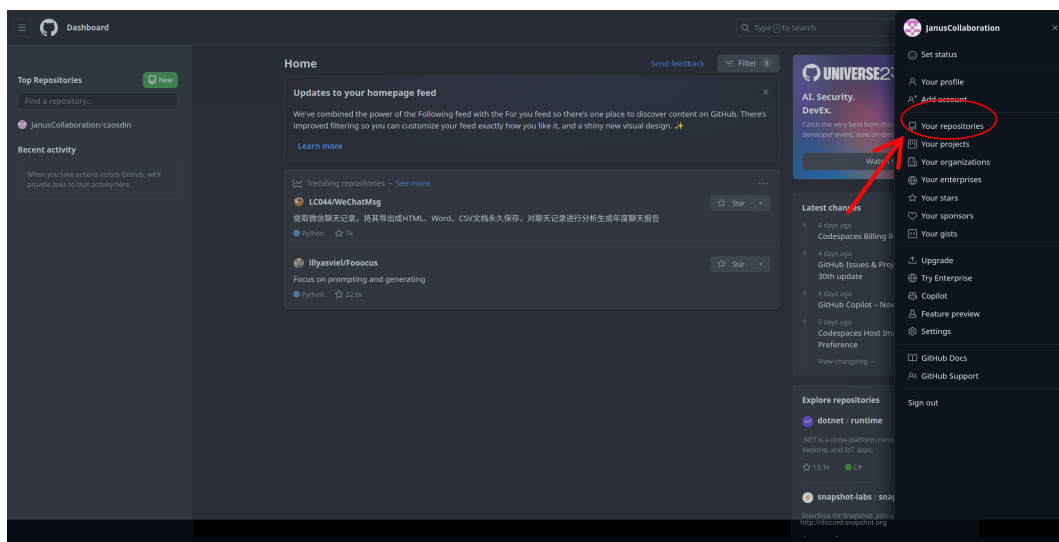


Figure 3: Selección de repositorios.

3 Paso 3. Usar repositorio.

Una vez creado el repositorio, nos saldrá una imagen parecida a (Fig. 6). Aquí se pueden añadir ficheros o directorios al repositorio.

Tal y como ocurre con la versión original (git) hay un sistema de ramas para permitir a un grupo grande de personas trabajar en paralelo, con un control de versiones muy sólido. Hice un minitutorial en su momento de git, os dejo el pdf en el propio repositorio de github.

Cuando añadimos ficheros o directorios, veis que podemos elegir si subirlos a la rama principal o a una nueva rama. En nuestro caso no creo que debamos preocuparnos mucho por eso, porque no habrá más de una persona realizando cambios (Fig. 7).

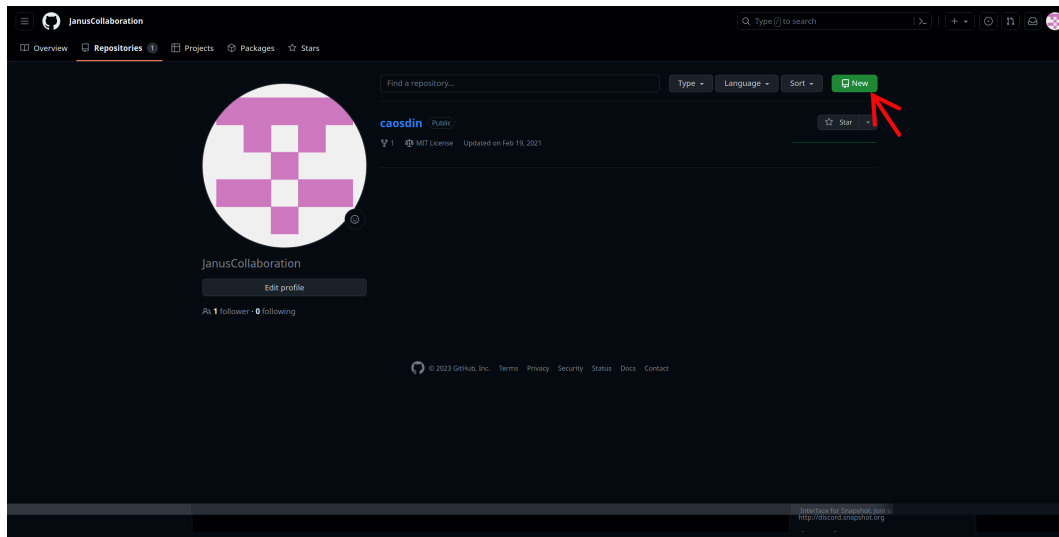


Figure 4: Página de repositorios.

Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository](#).

Required fields are marked with an asterisk ().*

Owner * JanusCollaboration

Repository name *

Great repository names are short and memorable. Need inspiration? How about **fuzzy-chainsaw** ?

Description (optional)

☒ **Public**
Anyone on the internet can see this repository. You choose who can commit.

☐ **Private**
You choose who can see and commit to this repository.

Initialize this repository with:

☒ **Add a README file**
This is where you can write a long description for your project. [Learn more about READMEs](#).

Add .gitignore
.gitignore template: None

Choose which files not to track from a list of templates. [Learn more about ignoring files](#).

Choose a license
License: MIT License

A license tells others what they can and can't do with your code. [Learn more about licenses](#).

This will set `main` as the default branch. Change the default name in your [settings](#).

ⓘ You are creating a public repository in your personal account.

[Create repository](#)

Figure 5: Creación de nuevo repositorio.

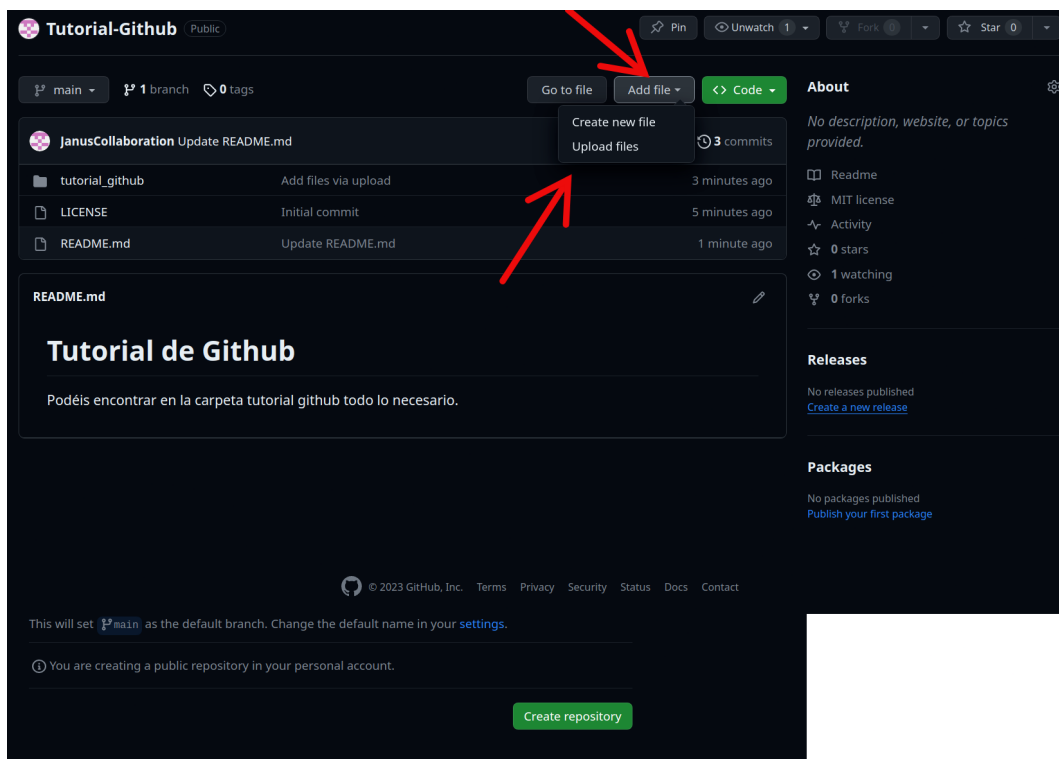


Figure 6: Añadir ficheros a un repositorio.

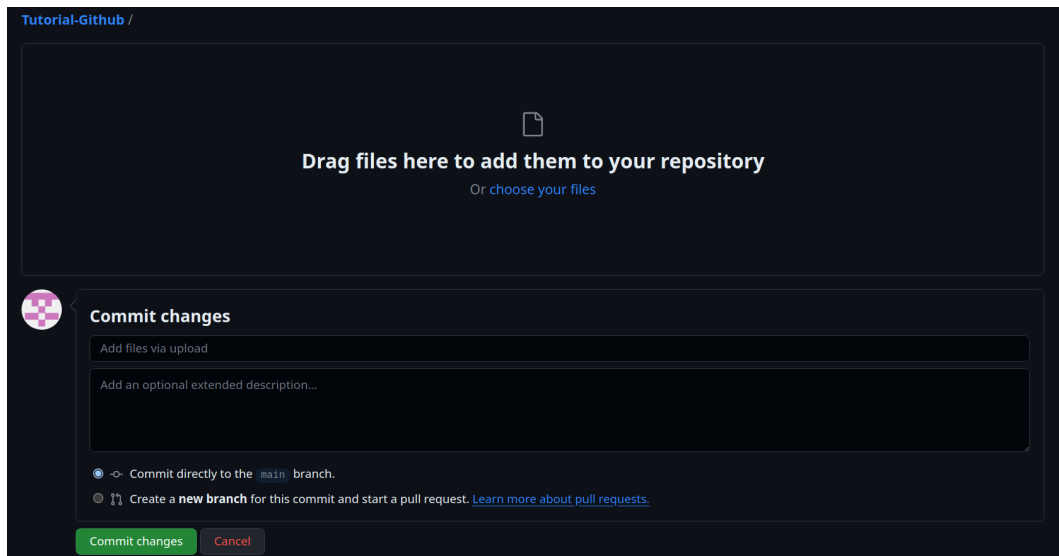


Figure 7: Interfaz de subida de fichero y manejo de ramas.